

What is claimed is:

1. A method for finding a local extrema for a processing element having a set of values associated therewith, the method comprising:
 - separating said set of values into an odd numbered set and an even numbered set;
 - determining an odd extrema from said odd numbered set;
 - determining an even extrema from said even numbered set; and
 - determining said local extrema from said odd extrema and said even extrema.
2. The method of claim 1 wherein said separating said set of values into an odd numbered set and an even numbered set comprises:
 - loading a value from an odd position within said set into a first register;
 - loading a value from an even position within said set into a second register and
 - transferring said value in said first register to a third register;
 - loading a value from a next odd position within said set into said first register and
 - transferring said value in said second register to a fourth register; and
 - loading a value from a next even position within said set into said second register.
3. The method of claim 2 wherein said determining an odd extrema from said odd numbered set comprises comparing the value in said first register to the value in said third register.
4. The method of claim 1 wherein said determining further comprises:
 - selecting the greater value from said first register and said third register if a high odd extrema is desired; and
 - selecting the lesser value from said first register and said third register if a low odd extrema is desired.
5. The method of claim 1 wherein said determining an even extrema from said even set of values comprises comparing the value in said second register to the value in said fourth register.
6. The method of claim 5 wherein said determining further comprises:
 - selecting the greater value from said second register and said fourth register if a high even extrema is desired; and
 - selecting the lesser value from said second register and said fourth register if a low even extrema is desired.

7. The method of claim 1 wherein said determining said local extrema from said odd extrema and said even extrema further comprises:
selecting the greater value from said odd extrema and said even extrema if a local high extrema is desired; and
selecting the lesser value from said odd extrema and said even extrema if the local low extrema is desired.
8. The method of claim 1 further comprising:
storing said odd extrema in said third register;
loading another value from an odd position within said set into said first register;
comparing the value in said first register to the value in said third register; and
repeating said storing, loading and comparing steps for remaining values within an odd position within said set.
9. The method of claim 5 further comprising:
storing said even extrema in said fourth register;
loading another value from an even position within said set into said second register;
comparing the value in said second register to the value in said fourth register;
and
repeating said storing, loading and comparing steps for remaining values within an even position within said set.
10. The method of claim 1 wherein said separating said set of values into an odd numbered set and an even numbered set comprises:
loading a first portion of an odd numbered value within said set into a first register;
transferring said first portion of said odd numbered value from said first register into a second register and loading a second portion of said odd numbered value into said first register;
transferring said second portion of said odd numbered value from said first register into a third register and loading a first portion of an even numbered value from within said set into said first register;

transferring said first portion of said even numbered value from said first register into a fourth register and loading a second portion of said even numbered value into said first register;

transferring said second portion of said even numbered value from said first register into a fifth register.

11. The method of claim 1 wherein said separating said set of values into an odd numbered set and an even numbered set comprises:

loading said least significant byte of a short value from an odd position within said set into a first register;

transferring said least significant byte of a short from an odd position in said first register to a second register and loading said most significant byte of said short value from said odd position within said set into said first register;

transferring said most significant byte of a short from an odd position in said first register to a third register, loading said least significant byte of a short value from an even position within said set into said first register, and initializing a fourth register with said least significant byte of a short from an odd position in said second register;

transferring said least significant byte of a short value from an even position within said set from said first register to a sixth register, loading said most significant byte of said short value from said even position within said set into said first register, and initializing a fifth register with said most significant byte of a short from said odd position in said third register;

transferring said most significant byte of said short value from said even position within said set from said first register to a seventh register, loading a least significant byte of a short value from another odd position within said set into said first register, and initializing a eighth register with said least significant byte of a short from said even position in said sixth register;

transferring said least significant byte of said short value from said another odd position within said set from said first register to said second register, loading a most significant byte of said short value from said another odd position within said set into said first register, and initializing a ninth register with said most significant byte of a short from said even position in said seventh register;

transferring said most significant byte of said short value from said another odd position within said set from said first register to said third register, and loading a least significant byte of a short value from another even position within said set into said first register;

transferring said least significant byte of said short value from said another even position within said set from said first register to said sixth register, and loading a most significant byte of said short value from said another even position within said set into said first register; and

transferring said most significant byte of said short value from said another even position within said set from said first register to said seventh register.

12. The method of claim 11 wherein said determining an odd extrema from said odd numbered set comprises comparing said least significant byte of said short value from said odd position within said set to said least significant byte of said short value from said another odd position within said set and comparing said most significant byte of said short value from said odd position within said set to said most significant byte of said short value from said another odd position within said set.

13. The method of claim 11 wherein said determining an even extrema from said even numbered set comprises comparing said least significant byte of said short value from said even position within said set to said least significant byte of said short value from said another even position within said set and comparing said most significant byte of said short value from said even position within said set to said most significant byte of said short value from said another even position within said set.

14. A method comprising:

identifying bytes of data within a data stream as having one of an odd or an even position;
processing said bytes of data having an odd position to produce an odd extrema;
processing said bytes of data having an even position to produce an even extrema;
and
determining a local extrema from said odd extrema and said even extrema.

15. The method of claim 14 wherein said processing said bytes of data having an odd position and processing said bytes of data having an even position comprises:

loading a byte of data having an odd position into a first register;
loading a byte of data having an even position into a second register and
transferring said byte of data in said first register into a third register;
loading a byte from a next odd position within said data stream into said first register and transferring said byte of data in said second register into a fourth register;

comparing said byte of data in said first register to said byte of data in said third register to produce said odd extrema and loading a byte of data from a next even position within said data stream into said second register; and

comparing said byte of data in said second register to said byte of data in said fourth register to produce said even extrema.

16. The method of claim 15 wherein said processing said bytes of data having an odd position to produce an odd extrema further comprises:

selecting the greater valued byte from said first register and said third register if a high odd extrema is desired; and

selecting the lesser valued byte from said first register and said third register if a low odd extrema is desired.

17. The method of claim 15 wherein said processing said bytes of data having an even position to produce an even extrema further comprises:

selecting the greater valued byte from said second register and said fourth register if a high even extrema is desired; and

selecting the lesser valued byte from said second register and said fourth register if a low even extrema is desired.

18. The method of claim 15 wherein said determining a local extrema from said odd extrema and said even extrema further comprises:

selecting the greater valued byte from said odd extrema and said even extrema if a local high extrema is desired; and

selecting the lesser valued byte from said odd extrema and said even extrema if the local low extrema is desired.

19. The method of claim 15 further comprising:

storing said odd extrema in said third register;

loading another byte from an odd position within said data stream into said first register;

comparing the byte within said first register to the byte within said third register;

and

repeating said storing, loading and comparing steps for remaining bytes within an odd position within said data stream.

20. The method of claim 15 further comprising:
storing said even extrema in said fourth register;
loading another byte from an even position within said data stream into said second register;
comparing the byte within said second register to the byte within said fourth register; and
repeating said storing, loading and comparing steps for remaining bytes within an even position within said data stream.
21. A method for determining a local extrema for a processing element, comprising:
loading odd numbered bytes of data into a first plurality of registers;
loading even numbered bytes of data into a second plurality of registers;
comparing certain of said loaded odd numbered bytes to produce an odd extrema;
comparing certain of said loaded even numbered bytes to produce an even extrema; and
producing a local extrema in response to said odd extrema and said even extrema.
22. The method of claim 21 wherein said loading odd numbered bytes of data into a first plurality of registers comprises:
loading a byte of data having an odd position into a first register;
transferring said byte of data in said first register into a third register; and
loading a byte from a next odd position within said data stream into said first register.
23. The method of claim 21 wherein said loading even numbered bytes of data into a second plurality of registers comprises:
loading a byte of data having an even position into a second register;
transferring said byte of data in said second register into a fourth register; and
loading a byte of data from a next even position within said data stream into said second register.
24. The method of claim 22 wherein said comparing certain of said loaded odd numbered bytes to produce an odd extrema comprises comparing said byte of data in said first register to said byte of data in said third register to produce said odd extrema.

25. The method of claim 23 wherein said comparing certain of said loaded even numbered bytes to produce an even extrema comprises comparing said byte of data in said second register to said byte of data in said fourth register to produce said even extrema.
26. The method of claim 24 wherein said comparing certain of said loaded odd numbered bytes to produce an odd extrema further comprises:
- storing said odd extrema in said third register;
 - loading another byte from an odd position within said data stream into said first register;
 - comparing the byte within said first register to the byte within said third register;
 - and
 - repeating said storing, loading and comparing steps for remaining bytes within an odd position within said data stream.
27. The method of claim 25 wherein said comparing certain of said loaded even numbered bytes to produce an even extrema further comprises:
- storing said even extrema in said fourth register;
 - loading another byte from an even position within said data stream into said second register;
 - comparing the byte within said second register to the byte within said fourth register; and
 - repeating said storing, loading and comparing steps for remaining bytes within an even position within said data stream.
28. The method of claim 14 wherein said processing said bytes of data having an odd position and processing said bytes of data having an even position comprises:
- loading said least significant byte of a short value from an odd position within said set into a first register;
 - transferring said least significant byte of a short from an odd position in said first register to a second register and loading said most significant byte of said short value from said odd position within said set into said first register;
 - transferring said most significant byte of a short from an odd position in said first register to a third register, loading said least significant byte of a short value from an even position within said set into said first register, and initializing a fourth register with said least significant byte of a short from an odd position in said second register;

transferring said least significant byte of a short value from an even position within said set from said first register to a sixth register, loading said most significant byte of said short value from said even position within said set into said first register, and initializing a fifth register with said most significant byte of a short from said odd position in said third register;

transferring said most significant byte of said short value from said even position within said set from said first register to a seventh register, loading a least significant byte of a short value from another odd position within said set into said first register, and initializing a eighth register with said least significant byte of a short from said even position in said sixth register;

transferring said least significant byte of said short value from said another odd position within said set from said first register to said second register, loading a most significant byte of said short value from said another odd position within said set into said first register, and initializing a ninth register with said most significant byte of a short from said even position in said seventh register;

transferring said most significant byte of said short value from said another odd position within said set from said first register to said third register, and loading a least significant byte of a short value from another even position within said set into said first register;

transferring said least significant byte of said short value from said another even position within said set from said first register to said sixth register, and loading a most significant byte of said short value from said another even position within said set into said first register; and

transferring said most significant byte of said short value from said another even position within said set from said first register to said seventh register.

29. The method of claim 28 wherein said determining an odd extrema from said odd numbered set comprises comparing said least significant byte of said short value from said odd position within said set to said least significant byte of said short value from said another odd position within said set and comparing said most significant byte of said short value from said odd position within said set to said most significant byte of said short value from said another odd position within said set.

30. The method of claim 28 wherein said determining an even extrema from said even numbered set comprises comparing said least significant byte of said short value from said even position within said set to said least significant byte of said short value from said another

even position within said set and comparing said most significant byte of said short value from said even position within said set to said most significant byte of said short value from said another even position within said set.

31. A memory device carrying a set of instructions which, when executed, perform a method comprising:
- separating said set of values into an odd numbered set and an even numbered set;
 - determining an odd extrema from said odd numbered set;
 - determining an even extrema from said even numbered set; and
 - determining said local extrema from said odd extrema and said even extrema.